A method and apparatus for managing a firewall

Patent Number: EP1024627

Publication date: 2000-08-02

Inventor(s): MAYER ALAIN JULES (US); BARTAL YAIR (US); WOOL AVISHAI

Applicant(s): LUCENT TECHNOLOGIES INC (US)

Requested Patent: JP2000253066

Application EP20000300371 20000119

Priority Number(s): US19990240934 19990129

IPC Classification: H04L12/24; H04L29/06; H04L12/22

EC Classification: H04L12/24, H04L29/06C6A

Equivalents:

Abstract

A method and apparatus are disclosed for managing a firewall. The disclosed firewall manager facilitates the generation of a security policy for a particular network environment, and automatically generates the firewall-specific configuration files from the security policy simultaneously for multiple gateways. The security policy is separated from the vendor-specific rule syntax and semantics and from the actual network topology. Thus, the security administrator can focus on designing an appropriate policy without worrying about firewall rule complexity, rule ordering, and other low-level configuration issues. In addition, the administrator can maintain a consistent policy in the presence of intranet topology changes. The disclosed firewall manager utilizes a model definition language (MDL) and an associated parser to produce an entity relationship model. A model compiler translates the entity-relationship model into the appropriate firewall configuration files. The entity-relationship model provides a framework for representing both the firewall-independent security policy, and the network topology. The security policy is expressed in terms of "roles," which are used to define network capabilities of sending and receiving services. A role may be assumed by different hosts or host-groups in the network. A visualization and debugging tool is provided to transform the firewall-specific configuration files into a graphical representation of the current policy on the actual topology, allowing the viability of a chosen policy to be evaluated. A role-group may be closed to prevent the inheritance of roles.

(19)日本国特許庁（ＪＰ）　(12)公開特許公報（Ａ）　(11)特許出願公開番号
特開2000－253066
（P2000－253066A）
(43)公開日　平成12年９月14日(2000.9.14)

(51)Int.Cl.'		識別記号	ＦＩ		テーマコード'（参考）
Ｈ０４Ｌ	12/56		Ｈ０４Ｌ	11/20	１０２Ｚ
Ｇ０６Ｆ	13/00	３５１	Ｇ０６Ｆ	13/00	３５１Ｚ
Ｈ０４Ｌ	12/46		Ｈ０４Ｌ	11/00	３１０Ｃ
	12/28			11/08	
	12/24			11/20	Ｂ

審査請求　未請求　請求項の数38　ＯＬ　（全 19 頁）　最終頁に続く

(21)出願番号　特願2000－19884(P2000－19884)

(22)出願日　平成12年１月28日(2000.1.28)

(31)優先権主張番号　０９／２４０９３４

(32)優先日　平成11年１月29日(1999.1.29)

(33)優先権主張国　米国（ＵＳ）

(71)出願人　596092698
ルーセント　テクノロジーズ　インコーポレーテッド
アメリカ合衆国．07974－0636　ニュージャーシィ，マレイ　ヒル，マウンテン　アヴェニュー　600

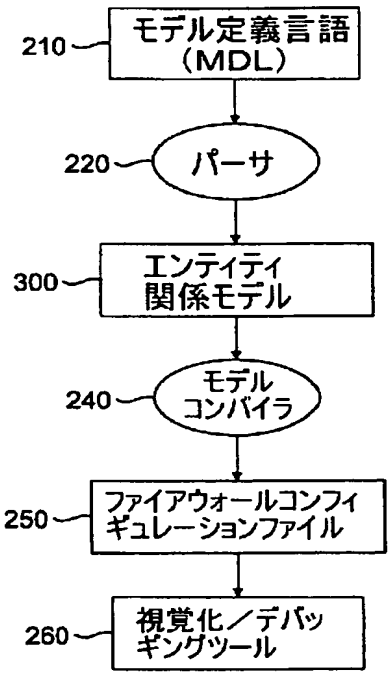
(72)発明者　ヤイル　バルタル
アメリカ合衆国　10012　ニューヨーク，ニューヨーク，モット　ストリート　284　アパートメント　ピーエッチシー

(74)代理人　100064447
弁理士　岡部　正夫　（外11名）

最終頁に続く

(54)【発明の名称】　ファイアウォールを管理するための方法および装置

(57)【要約】　（修正有）
【課題】ファイアウォールを管理する方法と装置を提供する。
【解決手段】ファイアウォールマネージャ２００は、モデル定義言語（MDL）２１０および関連するパーサ２２０を用いて、エンティティ関係モデル３００を生成する。モデルコンパイラ２４０は、エンティティ関係モデル３００を、適当なファイアウォールコンフィギュレーションファイル２５０に翻訳する。エンティティ関係モデル３００は、ファイアウォールに独立なセキュリティポリシーと網トポロジーの両方を表現する枠組を提供する。セキュリティポリシーは、“役割”の観点から表現される。役割は、サービスを送受信する網の機能を定義するために用いられ、網内の様々なホストあるいはホストグループに割当てられる。



【特許請求の範囲】

【請求項１】　複数のホストを含む網内の少なくとも一つのファイアウォールに対するコンフィギュレーションファイルを生成する方法であって、
ホストがパケットを送信および受信する能力を指定する複数の役割に対する定義を受信するステップ；前記役割の前記網内の前記複数のホストへの割当てを受信するステップ；および前記指定された役割に基づいて前記ホストに対する規則を生成するステップであって、前記規則がパケットが宛先ホストに通過できるか否かを決定するものであるステップを含むことを特徴とする方法。

【請求項２】　前記コンフィギュレーションファイルが前記網内の複数のファイアウォールに対して生成されることを特徴とする請求項１の方法。

【請求項３】　前記網に対するセキュリティポリシーが、網がサービスを送信および受信する能力を定義する前記役割の観点から表現されることを特徴とする請求項１の方法。

【請求項４】　前記複数のロールを結合することで、さまざまなロールグループが生成され、これらロールグループが一つあるいは複数のホストに割当てられることを特徴とする請求項１の方法。

【請求項５】　前記複数のホストを結合することで、ホストグループが形成され、ロールあるいはロールグループが、このホストグループに指定されることを特徴とする請求項１の方法。

【請求項６】　さらに、前記網内の前記複数のホストの構造の視覚表現を提供するステップを含むことを特徴とする請求項１の方法。

【請求項７】　さらに、前記コンフィギュレーションファイル内のセットの規則の視覚表現を提供するステップを含むことを特徴とする請求項１の方法。

【請求項８】　前記規則を生成するステップが、ベンダスペシフィックなコンパイラによって遂行され、ベンダスペシフィックなファイアウォールコンフィギュレーションファイルが生成されることを特徴とする請求項１の方法。

【請求項９】　複数の相互接続されたホストを含む網内の少なくとも一つのファイアウォールに対するコンフィギュレーションファイルを生成する方法であって、この方法が：モデル定義言語（MDL）を用いて、前記網に対するセキュリティポリシーを表現するエンティティ関係モデルを生成するステップ；および前記エンティティ関係モデルを前記ファイアウォールコンフィギュレーションファイルに翻訳するステップから、構成されることを特徴とする方法。

【請求項１０】　コンフィギュレーションファイルが前記網内の複数のファイアウォールに対して生成されることを特徴とする請求項９の方法。

【請求項１１】　前記セキュリティポリシーが、網のサービスを送信および受信する能力を定義する複数の役割の観点から表現されることを特徴とする請求項９の方法。

【請求項１２】　前記役割が前記ホストに割当られることを特徴とする請求項１１の方法。

【請求項１３】　前記複数のロールを結合することで、ロールグループが生成され、これらロールグループがホストに割当られることを特徴とする請求項１１の方法。

【請求項１４】　前記複数のホストを結合することで、ホストグループが形成され、ロールあるいはロールグループが、このホストグループに指定されることを特徴とする請求項１１の方法。

【請求項１５】　さらに、前記網内の前記複数のホストの構造の視覚表現を提供するステップを含むことを特徴とする請求項９の方法。

【請求項１６】　さらに、前記コンフィギュレーションファイル内のセットの規則の視覚表現を提供するステップを含むことを特徴とする請求項９の方法。

【請求項１７】　ベンダスペシフィックコンパイラが、前記エンティティ関係モデルをベンダスペシフィックなファイアウォールコンフィギュレーションファイルに翻訳することを特徴とする請求項９の方法。

【請求項１８】　複数のホストを含む網に対するセキュリティポリシーを表現するエンティティ関係モデルを生成する方法であって、
さらに許されるサービスとサービスが実行される方向を定義する一つあるいは複数のロールエンティティに対する定義を受信するステップ；前記網を、各隣接ゾーンに対するゲートウェイインタフェースを備えた一つあるいは複数のゲートウェイによって接続された、一つあるいは複数のゾーンに分割する前記網のトポロジーのモデルを受信するステップ；前記複数のホストの一つあるいは複数のゾーンへの割当を受信するステップ；および前記受信された定義、モデルおよび割当からエンティティ関係モデルを生成するステップを含むことを特徴とする方法。

【請求項１９】　さらに、前記ロールを前記複数のホストに割り当てるステップを含むことを特徴とする請求項１８の方法。

【請求項２０】　さらに、セットの前記ロールエンティティから成る一つあるいは複数のロールグループエンティティを定義するステップを含むことを特徴とする請求項１８の方法。

【請求項２１】　さらに、前記エンティティ関係モデルをファイアウォールコンフィギュレーションファイルに翻訳するステップを含むことを特徴とする請求項１８の方法。

【請求項２２】　前記コンフィギュレーションファイルが、前記網内の複数のファイアウォールに対して生成されることを特徴とする請求項２１の方法。

【請求項２３】　前記セキュリティポリシーが、網のサービスを送信および受信する能力を定義する複数のロールの観点から表現されることを特徴とする請求項１８の方法。
【請求項２４】　複数の前記ロールエンティティが結合されることで、ロールグループが生成され、ロールグループがホストに割当られることを特徴とする請求項１８の方法。
【請求項２５】　複数の前記ホストが結合されることで、ホストグループが形成され、ロールあるいはロールグループエンティティが、このホストグループに指定されることを特徴とする請求項１８の方法。
【請求項２６】　さらに、前記網内の前記複数のホストの構造の視覚表現を提供するステップを含むことを特徴とする請求項１８の方法。
【請求項２７】　さらに、前記コンフィギュレーションファイル内のセットの規則の視覚表現を提供するステップを含むことを特徴とする請求項２１の方法。
【請求項２８】　ベンダスペシフィックコンパイラが、前記エンティティ関係モデルをベンダスペシフィックなファイアウォールコンフィギュレーションファイルに翻訳することを特徴とする請求項１８の方法。
【請求項２９】　複数のホストを含む網に対するセキュリティポリシーを生成する方法であって、
ホストがパケットを送信および受信する能力を指定する複数の役割に対する定義を受信するステップ；前記ロールの前記網内の前記複数のホストへの割当を受信するステップ；および前記指定された定義および割当から前記セキュリティポリシーを生成するステップを含むことを特徴とする方法。
【請求項３０】　さらに、前記セキュリティポリシーを、前記網上のファイアウォールに対する少なくとも一つのコンフィギュレーションファイルに翻訳するステップを含むことを特徴とする請求項２９の方法。
【請求項３１】　前記コンフィギュレーションファイルが前記網内の複数のファイアウォールに対して生成されることを特徴とする請求項３０の方法。
【請求項３２】　複数の前記ロールを結合することで、ロールグループが生成され、ロールグループがホストに割当られることを特徴とする請求項２９の方法。
【請求項３３】　複数の前記ホストを結合することで、ホストグループが形成され、ロールあるいはロールグループが、このホストグループに指定されることを特徴とする請求項２９の方法。
【請求項３４】　さらに、前記網内の前記複数のホストの構造の視覚表現を提供する（を視覚的に表示する）ステップを含むことを特徴とする請求項２９の方法。
【請求項３５】　複数のホストを含む網内のファイアウォールに対するコンフィギュレーションファイルを生成するためのコンパイラであって、
コンピュータにて読むことができるコードを格納するためのメモリ；および前記メモリに結合されたプロセッサを備え、このプロセッサが前記コンピュータリーダブルコードを実行するように構成され、前記コンピュータリーダブルコードが、前記プロセッサを：ホストがパケットを送信および受信する能力を指定する複数の役割に対する定義を受信し；前記ロールの前記網内の前記複数のホストへの割当を受信し；前記指定されたロールに基づいて前記ホストに対する規則を生成するように構成され、前記規則がパケットが宛先ホストに通過できるか否かを決定（指定）することを特徴とするコンパイラ。
【請求項３６】　複数の相互接続されたホストを含む網内のファイアウォールに対するコンフィギュレーションファイルを生成するための：モデル定義言語（MDL）を用いて前記網に対するセキュリティポリシーを表現するエンティティ関係モデルを生成するパーサ；および前記エンティティ関係モデルを前記ファイアウォールコンフィギュレーションファイルに翻訳するためのコンパイラから、構成されるファイアウォールマネージャ。
【請求項３７】　複数のホストを含む網に対するセキュリティポリシーを表現するエンティティ関係モデルを生成するためのパーサであって、
コンピュータリーダブルコードを格納するためのメモリ；および前記メモリに結合されたプロセッサを備え、このプロセッサが前記コンピュータリーダブルコードを実行するように構成され、前記コンピュータリーダブルコードが、前記プロセッサを：さらに許されるサービスとサービスが実行される方向を定義する一つあるいは複数のロールエンティティに対する定義を受信し；前記網を、各隣接ゾーンに対するゲートウェイインタフェースを備えた一つあるいは複数のゲートウェイによって接続された、一つあるいは複数のゾーンに分割する前記網のトポロジーのモデルを受信し；前記複数のホストの一つあるいは複数のゾーンへの割当を受信し；前記受信された定義、モデルおよび指定からエンティティ関係モデルを生成するように、構成されることを特徴とするパーサ。
【請求項３８】　複数のホストを含む網に対するセキュリティポリシーを生成するためのシステムであって、
コンピュータリーダブルコードを格納するためのメモリ；および前記メモリに結合されたプロセッサを備え、このプロセッサが前記コンピュータリーダブルコードを実行するように構成され、前記コンピュータリーダブルコードが、前記プロセッサを：ホストがパケットを送信および受信する能力を指定する複数の役割に対する定義を受信し；前記ロールの前記網内の前記複数のホストへの割当を受信し；前記受信された定義および指定から前記セキュリティポリシーを生成するように構成されることを特徴とするシステム。
【発明の詳細な説明】

【0001】
【発明の属する技術分野】本発明は、一般的には、ファイアウォール、より詳細には、ファイアウォールを管理するための方法および装置に関する。
【0002】
【従来の技術】網ファイアウォールは、インターネットに接続されたあらゆる網に対する重要なセーフガード（防御手段）を提供する。ファイアウォールは、“アウトオブボックスに（out of box：箱から取り出すように）”起動できる単純なアプリケーションではなく、ファイアウォールは、ある与えられた会社あるいはエンティティの特定のニーズに対する重要なセキュリティポリシーを実現できるように構成（コンフィギァ）および管理する必要がある。ファイアウォールのセキュリティに影響を与える最も重要な要素は、ファイアウォールのコンフィギュレーション（構成）であると言われている。ファイアウォールは、強力な印象を与える技術的進歩ではあるが、ファイアウォールのコンフィギュレーションと管理の面では、今日まで殆ど進歩が見られなかったと言っても過言でない。
【0003】ファイアウォールは、パケットをフィルタリングし、プロプライアタリな（独自な）企業網、例えば、イントラネットと、公衆網、例えば、インターネットとを分離する網インタフェースである。今日の殆どのファイアウォールは、規則（ルール）ベースあるいはファイアウォールコンフィギュレーションファイルを用いて、構成（コンフィギャ）される。単一で、均質なイントラネット、例えば、小さな会社のローカルエリア網（LAN）を防御するファイアウォールの場合は、単一の規則ベースが、ファイアウォールに対して、どのような入り方向セッション（パケット）を通過させ、どのような入り方向セッションは遮断すべきかを指令する。同様に、この規則ベースは、どのような出方向セッション（パケット）は許可されるかも指定する。ファイアウォール管理者は、この下位の規則ベースを用いて、上位の企業セキュリティポリシーを実現することを必要とされる。
【0004】
【発明が解決しようとする課題】ファイアウォールのコンフィギュレーションインタフェースは、典型的には、セキュリティ管理者が、様々なホストグループ（IPアドレスのレンジ）とサービスグループ（プロトコルのグループと、エンドポイントを構成するホストの所の対応するポート番号）を定義するのを助ける。単一の規則（ルール）は、典型的には、ソース、宛先、サービスグループ、および適当な動作（アクション／処置）から構成される。ソースおよび宛先は、ホストグループに対応し、動作（アクション／処置）は、通常は、対応するセッションのパケットを、“通過（pass）”させるか、“脱落（drop）”させるかの指標に対応する。多くのファイアウォールにおいては、規則ベースは、順序に敏感である。換言すれば、ファイアウォールは、最初に、その規則ベース内の第一の規則が、新たなセッションに適用するか否かチェックし、第一の規則が適用する場合は、それらパケットは、第一の規則によって指定される動作（アクション／処置）に従って通過あるいは脱落（遮断）される。適用しない場合は、ファイアウォールは、第二の規則が適用するかチェックし、こうして、規則が順番にチェックされる。このスキームは、しばしば、規則ベース内の冗長な規則（リダンダントな規則）のために、不正確なコンフィギュレーションを与え、所望するセキュリティポリシーを実現するためには、これら規則の幾つかの順序を変えることが必要となる。
【0005】もう一つの可能なコンフィギュレーションエラーは、規則（ルール）がファイアウォールゲートウェイを完全に遮断するように設定されてしまい、新たな規則ベースがダウンロードできなくなるようなエラーである。いずれにしても、イントラネット全体のセキュリティは、規則ベースの具体的な内容に依存するが、現在は、より上位の抽象は利用できない。加えて、規則のシンタックスとセマンティクスおよび順番は、ファイアウォールの特定のメーク（作り）とモデルに依存する。
【0006】ファイアウォールの管理の問題は、一つのみでなく複数のファイアウォールを用いるより大きな会社ではより一層複雑となる。会社のイントラネットは複数のファイアウォールによって複数のゾーンに分割され、セキュリティポリシーは、典型的には、様々な異なるゾーンを互いに接続する複数のゲートウェイの所に配置された複数の規則ベースによって実現される。このため、様々な規則ベースの間の相互作用（インタプレイ）をセキュリティホールが発生しないように丹念に調べる必要があるが、規則ベースの設計および管理の複雑さは、イントラネットが複雑になればなるほど増加する。
【0007】ファイアウォールを管理する従来の技術に見られる上述の欠点から明らかなように、セキュリティポリシーを簡単に生成することができ、しかも、このセキュリティポリシーから、自動的に、規則ベースを、一つあるいは複数のゲートウェイに対して同時に生成することができるファイアウォールを管理するための方法および装置に対する要請が存在する。
【0008】
【課題を解決するための手段】本発明は、一般的には、ファイアウォールを管理するための方法および装置を開示する。開示されるファイアウォールマネージャは、特定の網環境に対するセキュリティポリシーを簡単に生成することができ、しかも、このセキュリティポリシーから、自動的に、個々のファイアウォールにスペシフィックな（適合された）コンフィギュレーションファイルを、複数のゲートウェイに対して同時に生成することができる。本発明の一面によると、セキュリティポリシー

は、ファイアウォール製造業者のスペシフィックな規則シンタックスおよびセマンティクスから分離され、セキュリティ管理者は、ファイアウォールの規則の複雑さ、規則の順番、および他の下位のコンフィギュレーション問題を気にすることなく、適当なセキュリティポリシーの設計に専念することが可能になる。

【０００９】本発明のもう一面によると、セキュリティポリシーは、実際の網トポロジーからも分離され、管理者は、イントラネットのトポロジーが変更された場合でも、一貫したポリシーを維持することが可能となる。さらに、このモジュール化により、管理者は、網の細部が異なる複数の企業サイトの所で同一のポリシーを再利用することが可能となり、他方、小さな会社では、エキスパート（専門家）によって設計されたデフォルトあるいは一例としてのセキュリティポリシーを用いることが可能となる。

【００１０】ファイアウォールマネージャは、モデル定義言語（MDL）および関連するパーサを用いて、エンティティ関係（リレーションシップ）モデルを生成する。モデルコンパイラは、エンティティ関係モデルを、適当なファイアウォールコンフィギュレーションファイルに翻訳する。エンティティ関係モデルは、ファイアウォールに独立なセキュリティポリシーと、網トポロジーの両方を表現する枠組（フレームワーク）を提供する。セキュリティポリシーは、“ロール（role、役割）”の観点から（を用いて）表現され、ロール（役割）は、網のサービスを送信および受信する能力（機能）を定義するために用いられる。ロールは、トポロジーとファイアウォールに独立なポリシーの本質を捉え、ロールは、網内の様々なホストによってとられる（に割当てられる）特性を表す。一群のロールをロールグループとして集合的に割当てることもできる。ホストグループあるいは個々のホストは、ロールグループが（属性を割当てられたロール（attribute assumed-roles）を介して）アタッチされるエンティティであり、従って、（ロールおよびロールグループの形でモデル化された）セキュリティポリシーが網トポロジーにリンクされる場所である。

【００１１】モデル定義言語（MDL）は、エンティティ関係モデルのインスタンスを定義するためのインタフェースとして用いられる。MDL言語に対するパーサは、エンティティ関係モデルのこれらインスタンスを生成する。モデルコンパイラは、モデルインスタンスを、ファイアウォールにスフィックな（適合された）コンフィギュレーションファイルに翻訳する。ファイアウォールにスペシフィックなコンフィギュレーションファイルを実際のトポロジー上の現在のポリシーのグラフ表現に変換するための視覚化／デバッギングツールが提供され、これによって、選択されたポリシーの実行可能性（妥当性）の評価が容易にされる。

【００１２】本発明のもう一面によると、ロールグループは、ロールの継承が適用されないように、クローズ（閉域）として定義される。クローズドロールグループを取る（割当てられた）ホストｈは、ｈを包含する任意の他のホストグループＡに割当てられた他のロール（役割）は継承しない。ホストには、最大でも、一つのクローズドロールグループが割当てられる。クローズドでないロールグループは、オープン（開放）であると呼ばれる。本発明のより完全な理解および本発明のその他の特徴および長所が、以下の詳細な説明および図面から得られるものである。

【００１３】

【発明の実施の形態】図１は、本発明による一例としての網の環境１００を示す。図１に示すように、網１００は、２つのファイアウォール（防火壁）１２０、１５０を備える。外側ファイアウォール１２０は、企業の外部網、例えば、インターネット１１０への接続を防御する。ファイアウォール１２０の内側には、サーバゾーン１３０が配置される。サーバゾーン１３０は、しばしば、“DMZ（demilitarized zone：非武装地帯／防壁セグメント）”とも呼ばれ、企業の外部からビジブルなサーバを含む。説明の実施例においては、サーバゾーン１３０内のビジブルなサーバは、電子メール（smtp:簡易メール転送プロトコル）ハイパーテキスト転送プロトコルファイル転送サービス（web）、およびファイル転送プロトコル（ftp）ファイル転送サービスを含むマルチプルサーバ１３８、並びにドメイン名サーバ（dns）サービス１３４から構成される。

【００１４】サーバゾーン１３０の内側には、イントラネットなどの企業のプロプライアタリな（独自な）、すなわち、内部の網を防御する内側ファイアウォール１５０が配置される。内側ファイアウォール１５０は、以下の３つのインタフェースを備える。第一は、サーバゾーン１３０へのインタフェースであり、第二のインタフェースは、内側ファイアウォール１５０を企業網ゾーン１６０に接続し、第三のインタフェースは、内側ファイアウォール１５０をファイアウォール管理ゾーン１４０に接続する。ファイアウォール管理ホストをセキュリティ化することは、網の保全性にとって非常に重要であり、このホストは、他の企業ホストとは分離されるべきである。企業網ゾーン１６０内には、通常は、制御ホストと呼ばれる一つの別個のホスト（図示せず）が設置され、これによって、サーバゾーン１３０内のサーバが管理される。説明の実施例においては、各ファイアウォール１２０、１５０は、おのおの、関連するパケットフィルタリングコンフィギュレーションファイル１２５、１５５を備える。これらについては、後に詳細に説明するが、概略的には、これらパケットフィルタリングコンフィギュレーションファイル１２５、１５５は、ファイアウォールに特定な（適合された）規則（ルール）ベースから成る。

【００１５】ファイアウォール管理ゾーン１４０は、後に図２との関連で詳細に説明するように、本発明によるファイアウォールマネージャ２００を備え、ファイアウォールマネージャ２００は、図１の一例としての網環境に対するセキュリティポリシー（方針）を生成する機能と、セキュリティポリシーからファイアウォールに特定な（適合された）コンフィギュレーションファイルを、複数のゲートウェイに対して同時に自動的に生成する機能を持つ。

【００１６】本発明の一つの特徴によると、セキュリティポリシーは、ファイアウォールの製造業者あるいはベンダの特定な規則シンタックス（構文）およびセマンティクス（意味）からは分離される。このため、本発明によると、セキュリティ管理者は、ファイアウォールの規則の複雑さ、規則の順序、および他の下位のコンフィギュレーション問題を気にすることなく、適当なポリシーを設計することに専念することが可能となる。加えて、セキュリティポリシーを、ベンダに特定な規則シンタックスおよびセマンティクスから分離することで、異なるベンダからの網要素を統一的に管理することができ、会社がベンダを変える際の切替えも楽になる。

【００１７】本発明のもう一つの特徴によると、セキュリティポリシーは、実際の網トポロジーからも分離される。このため、本発明によると、管理者は、イントラネットのトポロジーが変更された場合でも、一貫したセキュリティポリシーを維持することができる。さらに、このモジュール化のために、管理者は、網の細部が異なる複数の企業サイトにおいて、同一のセキュリティポリシーを再使用することができる他、小さな会社では、エクスパート（専門家）によって設計されたデフォルト（省略時）のあるいは一例としてのセキュリティポリシーを用いることができる。

【００１８】本発明は、コンピュータによって実現される方法を用いて、セキュリティポリシーからファイアウォールコンフィギュレーションファイルを、自動的に、複数のゲートウェイに対して同時に生成する。このため、ファイアウォールに特定なコンフィギュレーションファイル内に検出困難なエラーが導入される確率を低減することができる。加えて、本発明では、コンフィギュレーションファイルの上位のデバッギングが許されるとともに、セキュリティ管理者は、コンフィギュレーションファイル内の情報を容易に把握することができる。

【００１９】図２は、本発明によるファイアウォールマネージャ２００の様々な構成要素を示す。エンティティ関係（リレーションシップ）モデル３００は、図３との関連で後に詳細に説明するように、ファイアウォール独立なセキュリティポリシーと、網トポロジーの両方を表現するためのフレームワーク（枠組）を提供する。後に詳細に説明するように、セキュリティポリシーは、“ロール（役割）”の観点から表現され、ロールは、網の機能（能力）を定義するために用いられる。ロールは、網のトポロジーとファイアウォールの両方に独立なセキュリティポリシーの本質を捉える（を表す）。

【００２０】MDL（model definition language：モデル定義言語）２１０は、エンティティ関係モデル３００のインスタンスを定義するためのインタフェースとして用いられる。MDLに対するパーサ（解析器）２２０も開示されるが、これは、エンティティ関係モデル３００のインスタンスを生成するために用いられる。

【００２１】モデルコンパイラ２４０は、モデルのインスタンスを、一つあるいは複数のファイアウォールコンフィギュレーションファイル２５０に翻訳する。これらセットのファイアウォールコンフィギュレーションファイル２５０は、典型的には、網のトポロジーと規則ベースの情報を含む。現代の殆どのファイアウォールでは、フィルタリング機能を複数のゲートウェイに配布（分配）することができるようになっており、この場合は、コンパイラ２４０は、各ゲートウェイに対して別個のセットのローカルコンフィギュレーションファイルを生成することが必要となる。

【００２２】視覚化／デバッギングツール２６０は、ファイアウォールに特定な（適合された）コンフィギュレーションファイルを、実際の網トポロジー上の現在のセキュリティポリシーのグラフ表現に変換する。

【００２３】ファイアウォールの用語とモデリングの概念

ゲートウェイは、ここでは、パケットフィルタリングマシーンとして用いられ、ファイアウォールもしくはルータから成る。ゲートウェイは少なくとも２つのインターネット接続を持つために、ゲートウェイは、定義からして、マルチホーム（multi-homed）である。各接続は、インタフェースを通り、インタフェースは、自身の一意のIPアドレスを持つ。各インタフェースは、おのおの、関連するパケットフィルタリングコンフィギュレーションファイルを備えるものと想定される。ゲートウェイは、IPアドレス空間を、図１に示すように、互いに素な（共通要素を持たない）ゾーンに分割する。殆どのゾーンは、会社のサブネットに対応し、通常は、一つの大きな“インターネット（Internet）”ゾーン１１０がその企業によって用いられてないIPアドレス空間の部分に対応する（を構成する）。サービスは、tcp（Transmission Control Protocol）やudp（User Datagram Protocol）などのプロトコルベースと、発信側と宛先側の両方のポート番号の組合せとして定義される。例えば、telnetサービスは、宛先ポート２３と任意のソースポートを持つtcpとして定義される。

【００２４】ロール（役割）は、後に図３との関連で詳細に説明するように、網の様々な異なるホストによって引き受けられる（に割当てられる）特性を表す。より具体的には、ロールは、受信サービスと送信サービスの能

力を定義する。例えば、“メールサーバ（mail-server）”なるロールは、任意の箇所からメールサービスをと受信する能力を定義する。少し複雑なセキュリティポリシーとして、“メールサーバ（mail-server）”なるロールと、“内部メールサーバ（internal mail-server）”なるロールを導入することも考えられる。内部メールサーバなるロールは、メールサーバのロール（役割）を受け持つ（割当てられた）ピア（仲間）からのsmtp（簡易メール転送プロトコル）電子メールを受信する能力を定義する堅固な“sendmail（メール送信）”モジュールを備えたメールサーバが、メールサーバのロールを受け持ち（割当てられ）、イントラネットの内側の通常のメールサーバは、内部メールサーバのロール（役割）を受け持つ（を割当てられる）。ロールは、受信能力に加えて、送信能力を定義するためにも用いられる。アウトバンド（出方向）ロール、例えば、ウエブイネーブルド（起動）ロールは、外側世界へのハイパーテキスト転送プロトコル（http）通信を可能にする。アウトバンドロールは、典型的には、イントラネットのホストにアタッチされ、これを用いることで、従業員は、ウエブをブラウズすることが可能になる。

【００２５】ロールは、本質的には、（１）許されるサービス、および（２）そのサービスが適用するピアを定義する。ピアは、他の（あるいは同一の）ロールによって表現される。最終的には、ロールは、実際のホスト（マシーン）に割当てられ、こうして、セキュリティポリシー（ロール）が実際のトポロジーに結び付けられる。便宜的に、ロールグループ（ロールの集まり）も定義され、ホストに割当てられる。ロールは、ポジティブ（積極的）な能力を指定し、“whatever is not explicitly allowed is disallowed（明示的に許可されてないものは全て不許可）”なる戦略を、暗黙的（インプリシット）に実現する。例えば、あるホストは、http（ハイパーテキスト転送プロトコル）リクエストを、そのホストにウエブサーバの対応するロールが指定されている場合に限り受理（受信）する。

【００２６】網トポロジーは、後に図３との関連で詳細に説明するように、エンティティ関係モデル３００によってもモデル化される。ホストエンティティは、網上のマシーンを、自身のIPアドレスと名前にてモデル化する。ホストグループは、IPアドレスのレンジ、あるいは、セットの他のホストもしくはホストグループによって定義されるセットのマシーンを表す。インヘリタンス（継承）も、通常の方法にて用いられ、ホストｈが受け持つ（に割当てられた）セットのロールは、全て、ｈを包含するホストグループにも割当てられる（によっても継承される）。

【００２７】ペイロール（経理）用のサブネットをイントラネットの他の部分と接続するゲートウェイの場合は、そのゲートウェイへのアクセスが、小さなセットの他のマシーンからの限られた管理タスクに制限されることを保証することが重要であり、さもなければ、ルーティングおよびアクセス制御が、容易にコラプト（失墜）する（内部への不法な侵入を受ける）恐れがある。換言すれば、ゲートウェイは、“秘密化（stealthed）”すべきである。ゲートウェイに、非常に限定された機能のみを許可（定義）するロール、例えば、ホストからのリモート管理にファイアウォールドメインロールを指定することも考えられるが、ただし、この方法では、ゲートウェイが、望ましくないアクセスを許す他のロールを継承する恐れがある。

【００２８】このために、本発明のもう一面においては、クローズド（閉じられた／閉域）ロールグループが導入される。クローズドロールグループは、ロールの継承が適用しないようなロールグループである。つまり、クローズドロールグループを受け持つ（割当てられた）ホストｈは、ｈを包含する任意の他のホストグループＡに割当てられたロールは継承しない。ホストは、最高でも一つのみのクローズドロールグループを割当てられる。クローズドでないロールグループは、オープン（開放）であると呼ばれる。ロールグループは、デフォルトにより、オープンとすることができる。ここでも、ロールは、ポジティブ（積極的）な能力を表明し、このため、“クローズド（closed）”ロール機構は、限られた形式のネガティブ（消極的）な表現を提供する。

【００２９】エンティティ関係モデル

図３はエンティティ関係モデルの枠組を示す。図３において、単一の矢頭を持つ矢印は、一対一の関係を表し、二重の矢頭を持つ矢印は、一対多数の関係を表す。図３は、オブジェクト指向エンティティ関係モデルの枠組の階層を表す。本発明によると、エンティティ関係モデル３００は、ファイアウォールに独立なセキュリティポリシーと網トポロジーを表現するためのオブジェクト指向フレームワークを提供する。セキュリティポリシーは、一つあるいは複数の網能力（機能）３１５を定義する“ロール（role）”オブジェクトの観点から表現される。網内の各マシーンは、“ホスト（host）”オブジェクト３８０としてモデル化される。一般的には、ホストオブジェクト３８０あるいはホストグループオブジェクト（定義されたホストオブジェクトの集合）３７０にある与えられたロールオブジェクト３１０あるいはロールグループオブジェクト（定義されたロールの集合）３２５を割当てられた場合、ホストオブジェクト３８０あるいはホストグループオブジェクト３７０はロールオブジェクト３１０あるいはロールグループオブジェクト３２５に対して定義されている網能力を継承する。

【００３０】網トポロジーは、以下のようにモデル化される。すなわち、網は、ゾーンオブジェクト３４０に分割され、各ゾーンは、ゲートウェイオブジェクト３５０を介して接続される。各ゾーンオブジェクト３４０は、

一つあるいは複数のホストオブジェクト３８０から構成される。ゲートウェイ３５０は、隣接する各ゾーン３４０に対するゲートウェイインタフェースオブジェクト３６０を備える。ゾーン（オブジェクト）３４０に出入りするパケットは、対応するゲートウェイインタフェース（オブジェクト）３６０上のゲートウェイ（オブジェクト）３５０によってフィルタリングされる。終局的には、ロール（役割）オブジェクトは、実際のホストオブジェクト（マシーン）に割当てられ、こうして、セキュリティポリシー（ロール）が実際の網トポロジーに結び付けられる。

【００３１】図3に示すように、ロールエンティティ３１０は、一つあるいは複数のピア機能（能力）３１５から成る。各機能（能力）３１５は、その属性を介して、許可されるサービス３２０と、サービス３２０を実行することを許されるピア、および方向を定義する。換言すれば、サービスの方向は、サービス３２０が、出方向機能（能力）３１５としてロール３１０からピアの方向に実行されるか、入方向機能（能力）３１５としてピアからロール３１０の方向に実行されるかを示す。サービスエンティティ３２０は、プロトコルベースと、2つのポート番号属性、つまり、dest-port-no-range（宛先ポート番号レンジ）およびscr-port-no-range（ソースポート番号レンジ）を持つ。ピアは、もう一つの（あるいは同一の）ロール３１０をポイントする。

【００３２】ロールグループエンティティ３２５は、セットのロール３１０から構成される。ロールグループエンティティ３２５も、後により詳細に説明する、クローズと呼ばれるブール属性を持ち、クローズブール属性は、そのロールグループ３２５がクローズド（閉じた）ロールグループであるか否かを示す。再度、クローズドロールグループを指定されたホストは、他のロールは継承しないことに注意する。

【００３３】網トポロジーは、以下のようにモデル化される。つまり、網は、ゾーン３４０に分割され、ゲートウェイ３５０を介して接続される。ゲートウェイ３５０は、各隣接するゾーン３４０に対するゲートウェイインタフェース３６０を備え、各ゲートウェイインタフェース３６０は、（インタフェースホストの属性によってモデル化される）自身のIPアドレスを持つ。ゾーン３４０に出入りするパケットは、対応するゲートウェイインタフェース３６０上のゲートウェイ３５０によってフィルタリングされる。ただし、同一のゾーン３４０の内側で送受信されるパケットについては、これらはどのゲートウェイ３５０も通らないために、ゲートウェイ３５０によってフィルタリングされることはない。ゾーン３４０は、ホストグループ３７０から構成され、これらホストグループ３７０は、典型的には、さらに、より小さなホストグループ３７０もしくは単一のホスト３８０の階層にサブ分割される。各ホストグループ３７０は、自身と他のホストグループ３７０との関係、すなわち、コンテインメント（包含、containment）とインタセクション（共通、intersection）を、“contains（包含）”と“intersects（共通）”属性に格納する。

【００３４】ホストグループ３７０およびホスト３８０は、ロールグループ３２５が（属性割当ロール（attribute assumed-role）を介して）アタッチされる（割当てられる）エンティティである。つまり、ここは、（ロール３１０とロールグループ３２５によってモデル化される）セキュリティポリシーが網トポロジーにリンクされる（結び付けられる）場所である。

【００３５】MDL（Model Definition Language：モデル定義言語）

MDL（モデル定義言語）２１０は、セキュリティポリシーをインスタンス化し、セキュリティポリシーを網トポロジーにマッピングするために用いられる。パーサ２２０は、MDLプログラム２１０を、エンティティ関係モデル３００のインスタンスに翻訳する。モデル３００は、対応するデータ構造によって表現される。

【００３６】（１）セキュリティポリシーを記述するための言語

サービス３２０は、以下の形式のステートメントによって定義される：

【数１】

<service-name> =

<protocol-base> [<dest-port-no-range>, <src-port-no-range>].

例えば、以下のコードフラグメントは、広く使用されているサービスであるsmtp、ssh、ping、https、およびall_tcpベースのパケットを指定するサービスを定義する：

【数２】

```
SERVICES {
        smtp   =    TCP [25]
        ssh    =    TCP [22]
        ping   =    ICMP [8,0]
        https  =    TCP [443]
        all_tcp =   TCP [*]
}
```

サービスは、サービスグループ、すなわち、ServiceGrpに、以下の形式のステートメントを用いてグループ化される：

【数３】

<srv-grp-name> = {<service-name1>, <service-name2> ...}

【００３７】以下のコードフラグメントは、２つのサービスグループ、すなわち、admin-to-gtwy（ゲートウェイから管理ゾーン）と、gtwy-to-admin（管理ゾーンからゲートウェイ）を定義する：

【数４】

```
SERVICE_GROUPS {
        admin-to-gtwy = {ssh, ping}
        gtwy-to-admin = {ssh, https}
```

【００３８】ロール３１０は、以下の形式のステートメントによって定義され、ここで、矢印は、方向属性を明らかなやり方で定義し、role-grp-name（ロールグループ名）は、ピアをポイントし、ser-grp-name（サービスグループ名）は、サービスグループ３３０をポイントする：

【数５】

<role-name> arrow <role(-grp)-name> : <srv-grp-name>

arrow == ← || → || ↔

【００３９】以下のコードフラグメントは、上述のロール、すなわち、mail_server（メールサーバ）とinternal_mail_server（内部メールサーバ）を定義する。gateway-in（ゲートウェイ・イン）なるロールと、gateway-out（ゲートウェイ・アウト）なるロールは、ゲートウェイインタフェースの各方向における能力（機能）をモデル化する。これらのことが以下のコードフラグメントに例示される：

【数６】

```
ROLE_DEFINITIONS {
        mail_server ↔ * : smtp
        internal_mail_server ↔ mail_server : smtp
        gateway_in ← fw_admin : admin_to_gtwy
        gateway_out → fw_admin : gtwy_to_admin
        intranet_machine → all_tcp : *
}
```

【００４０】ロール３１０は、以下のステートメントにて、（デフォルトによる）オープンロールグループ３２５にグループ化される：

【数７】

<role-grp-name> = { <role-name1>, <role-name2> ...}

【００４１】ロール３１０は、以下のステートメントにてクローズドロールグループ３２５にグループ化され

る：

【数８】

<role-grp-name> = << <role-name1>, <role-name2> ... >>

【００４２】以下のコードフラグメントは、ロールグループ、ゲートウェイ、単方向ゲートウェイロール３１０の一つのロールグループ３２５へのバンドリングを定義する。このゲートウェイロールグループは、閉じられており、従って、このロールグループを受け持つ（割当てられた）ホストを、効果的に“stealthing（秘密化）”することに注意する：

【数９】

```
ROLE_GROUPS {

    gateway     =     <<gateway_in, gateway_out>> # a closed group
```

【００４３】（２）網トポロジーの記述およびセキュリティポリシーのマッピングのためのMDL（モデル定義言語）

ホスト３８０およびホストグループ３７０は、以下のステートメントによって定義される：

【数１０】

<host-name> = [<IP-Addr>] : <role-grp-name>

<host-grp-name> = [<IP-Range>] : <role-grp-name>

【００４４】以下のコードフラグメントは、ホストを、ダーティ（dirty）（恐らくはトントラネットの外側）と、ダスティ（dusty）に定義し、これらに、それぞれ、外部および内部メールサーバのロールを割当てる：

【数１１】

```
HOST {
    dirty  =   [ 111.222.100.6 ]   : mail_server
    dusty  =   [ 111.222.1.3]      : internal_mail_server
}
```

ゲートウェイ３５０は、以下のステートメントによって定義される：

【数１２】

<gateway-name> = { <host-name1>, <host-name2> ...}

【００４５】以下のコードフラグメントは、payroll_gw_interface1/2をホストとして定義し、それらのIP-アドレスを指定し、次に、payroll_gwなるゲートウェイを、自身の２つのインタフェースとして、payroll_gw_interface1/2を持つものとして定義する。このコードフラグメントは、さらに、ゲートウェイなるロールグループを、これら、インタフェースに割り当る。

【数１３】

```
HOST {
    payroll_gw_interface1 = [111.222.26.226 ] : gateway
    payroll_gw_interface2 = [111.222.24.210 ] : gateway
}

GATEWAYS {
    payroll_gw = { payroll_gw_interface1, payroll_gw_interface2}
}
```

ゾーン３４０は、以下のステートメントによって定義される：

【数１４】

<zone-name> : { <gtwy-interface-name1>, <gtwy-interface-name2> ... }

【００４６】以下のコードフラグメントは、最初に、（イントラネットmanhattan_officeの一部である）payroll_zoneとcorp_zoneなるゾーンをホストグループとして定義し、次に、これらのIP-レンジを指定し、次に、網のトポロジーの部分を定義する。後者の部分は、payroll_zoneはpayroll_gw_interface1によってpayroll_gwに接続され、payroll_gwの第二のインタフェースはcorp_zoneに接続されることを指定する：

【数１５】

```
HOST-GROUPS {
manhattan_office = [111.222.0.0-111.222.255.255] : intranet_machine
payroll_zone     = [111.222.26.0-111.222.26.255] : payroll_machine
corp_zone        =[111.222.24.0-111.222.24.255] :
non_payroll_machine
}
ZONES {
payroll_zone  = {payroll_gw_interface1}
corp_zone     = { payroll_gw_interface2, ... }
}
```

【００４７】モデルコンパイラ

セキュリティ管理者によって、セキュリティポリシーが、計画（設計）され、MDL（モデル定義言語）２１０にてプログラム化され、MDLパーサ２２０が、エンティティ関係モデル３００を生成した後に、エンティティ関係モデル３００は、モデルコンパイラ２４０によって、適当なファイアウォールコンフィギュレーションファイル２５０に翻訳される。この翻訳は、結果としてのファイルが、底辺に横たわるセキュリティポリシーを正しく実現することを保証する必要がある。コンフィギュレーションファイル２５０は、典型的には、サービスの定義、ホストグループの定義、および各ゲートウェイインタフェースに対するコンフィギュレーションファイルを含むために、コンパイラ２４０のバックエンドは、ベンダスペシフィックであることを要求される。

【００４８】サービスおよびホストグループに対するコンフィギュレーションファイル２５０は、当業者においては明らかな方法にて、率直なやり方で生成される。ここに説明の一般ファイアウォールは、順序付きリストを用い、明示的に許可されてないものは、全て不許可とする（拒絶する）。この一般規則フォーマットは、以下のフィールド、すなわち、sorce host-group（ソースホストグループ）、destination host-group（宛先ホストグループ）、service/service-group（サービス／サービスグループ）、action（動作）（例えば、pass/drop（通過／脱落））、およびdirection（方向）を含む。方向フィールドは、上述のロールエンティティ３１０の方向属性とは異なることに注意する。パケットがフィルタリングされるとき、リスト内の規則（ルール）が、リストの順に一致が見つかるまで調べられ、一致が見つかった時点で、対応する動作（action）が遂行される。リスト内の最後の規則は、デフォルト規則であり、これは、全てのパケットを脱落させる（拒絶する）。

【００４９】一つの実施例においては、ファイアウォールコンフィギュレーションファイル２５０の生成は、２つの部分に分けて行なわれる。モデルコンパイラ２４０は、図4および図5に示すように、中央（集中）ファイアウォールコンフィギュレーションファイル２５０Ａを生成するための基本モデルコンパライ４１０を備える。加えて、モデルコンパイラ２４０は、それぞれ、ゲートウェイインタフェース１２０、１５０に適合（アダプト）されるパケットフィルタリングコンフィギュレーションファイル１２５、１５５を生成するためのコンフィギュレーションファイルトポロジーアダプタ４２０を備える。

【００５０】基本モデルコンパイラ４１０は、エンティティ関係（リレーションシップ）モデル３００のインスタンスを、ファイアウォールコンフィギュレーションファイル２５０Ａに翻訳する。基本モデルコンパイラ４１０は、網の構造、例えば、ゲートウェイの位置は無視して、ロール３１０、ロールグループ３２５の定義、およびこれらのホストグループ３７０への割当てに専念する。基本モデルコンパイラ４１０は、ロール３１０およびロールグループ３２５の定義を用いて、どのペアのホストグループ３７０がこれらの間で行なわれる特定のサービスを許可するファイアウォール規則を持つべきかを、どのゲートウェイ１２０、１５０が実際にこの規則を執行（実現）することができるか否かの疑問は無視して、演繹的に類推する。基本モデルコンパイラ４１０の出力は、従って、セキュリティポリシーを実現するための要求される全てのの規則を含む単一の中央ファイアウォールコンフィギュレーションファイル２５０Ａとな

る。この中央ファイアウォールコンフィギュレーションファイル２５０Ａは、規則の方向フィールドは設定しない。上述のように、各規則の方向（の指定）は、コンフィギュレーションファイルトポロジーアダプタ４２０によって後段において達成される。

【００５１】ロール（役割）の定義は、どのような動作がそれらロールを割当てられたマシーンの間で許可されるかの記述である。このことは、ロールグループが特定のホストグループＨに割当てられた場合、そのホストグループＨと、ピアのロールを割当てられた他のホストは、セットのポジティブ（積極的）な規則を共有することを意味し、これらセットの規則は、Ｈと関連すると言われる。全てのロールグループがオープンである（開かれている）場合は、これらポジティブな規則は、衝突（矛盾）することはなく、従って、正しいコンフィギュレーションファイルを形成する。

【００５２】ただし、クローズドロールグループの扱いは、より複雑となる。例えば、ｈには、クローズドロールグループＣが割当てられており、Ｈは、ｈを包含するホストグループであり、しかも、Ｈには、異なるロールグループＲが割当てられている状況を想定する。ここで、ｈには、クローズドロールグループが割当てられているという事実は、ｈは、ホストグループＨからはどのようなロールも継承すべきでないことを暗に意味する。ところが、（Ｒによって含意される）Ｈと関連するセットの積極的な規則が生成された場合、幾つかのサービスは、ｈに対して不当に許可されてしまうこととなる。

【００５３】この問題は、基本モデルコンパイラ４１０によるホストグループの分割の際に、結果としてのホストグループがクローズドロールグループを割当られたホストは含まないようにすることで回避できる。例えば、仮に、基本モデルコンパイラ４１０が、ホストグループＨを、ｈを除く同一のホストグループから成るＨ’にて置き換え、このｈを除くホストグループＨ’にロールグループＲをＨ’に割り当てるようにした場合は、基本モデルコンパイラ４１０は、（Ｈではなく）Ｈ’と関連するセットの積極的な規則を生成し、結果として、積極的な規則のみを生成することとなる。ただし、この解決策は、ユーザ未定義なホストグループ（non user-defined host-groups）の生成はデバッギング過程をより困難なものとするために、最適な策とは言えない。

【００５４】このために、本発明においては、消極的（ネガティブ）な規則を用いることで、新たなホストグループの必要性が回避される。この方式では、直感的に、クローズドロールグループを扱うポジティブ（積極的）な規則が、コンフィギュレーションファイル内で、他の規則より前に現われ、これらポジティブ（積極的）な規則の後に、“そのホストグループに対しては指定されるもの以外は許可されない（nothing else is allowed for the host group）”なる概念を持つネガティブ（消極的）な規則が続くことを要求される。オープンロールグループのみを扱う規則は、全てのクローズドロールグループが処理された後に初めて出現するようにされる。ホストグループは、そのホストグループにクローズドロールグループが割当てられている場合は、クローズと呼ばれ、割当てられてない場合は、オープンと呼ばれる。

【００５５】図６は、基本モデルコンパイラ４１０によって遂行される一例としての規則生成アルゴリズム６００を示す。図６に示すように、規則生成アルゴリズム６００は、３つの過程から構成され、この過程が終了した時点で、デフォルト（省略時）のネガティブ（消極的）な規則がファイアウォールコンフィギュレーションファイル２５０に加えられる。

【００５６】最初に、過程１のステップ６１０において、クローズドホストグループの各ペアに対して、これらの間の全てのポジティブ（積極的）な規則が生成される。その後、ステップ６２０において過程１で生成されたポジティブな規則が中央ファイアウォールコンフィギュレーションファイル２５０Ａに挿入される。

【００５７】次に、過程２のステップ６３０において、クローズド（閉じた／閉域）ホストグループH_1と、オープン（開いた／開放）ホストグループH_2の各ペアに対して、これらの間の全てのポジティブな規則が生成される。次に、ステップ６４０において、オープンホストグループH_2内に含まれる各クローズドホストグループＧに対して、H_1とＧとの間の全てのネガティブ（消極的）な規則が生成される。その後、ステップ６５０において、過程２のネガティブな規則と、これに続く、過程２のポジティブな規則が中央ファイアウォールコンフィギュレーションファイル２５０Ａに挿入される。

【００５８】次に、過程３のステップ６６０（図６Ｂ）において、各クローズドホストグループＨに対して、Ｈとall-hosts（オールホスト）なるホストグループとの間のネガティブな規則が生成される。その後、ステップ６７０において、生成されたネガティブな規則が中央ファイアウォールコンフィギュレーションファイル２５０Ａに挿入される。次に、ステップ６８０において、オープンホストグループの各ペアに対して、これらの間の全てのポジティブな規則が生成される。次に、ステップ６９０において、これらポジティブな規則が中央ファイアウォールコンフィギュレーションファイル２５０Ａに挿入され、プログラム制御は終了する。

【００５９】規則生成アルゴリズム６００の基本要素として、ペアのホストグループH_1とH_2に対して、H_1と関連し、かつ、H_2にも適用するポジティブな規則のリストを生成する必要がある。図７は、これらポジティブな規則のリストを生成するためのルーチンを図解する。図７に示すように、これらポジティブな規則は、以下の疑似コードによって生成される：

【外１】

for each role r in the role-group assigned to H_1:

for each statement in the form: { r $ R : s}

if H_2 is closed: /* create a rule if H_2 has a role, r */

if the role-group assigned to H_2 contains a role in R, then

create a positive rule between H_1 and H_2 with service=s

otherwise, for all host-groups G that contain H_2:

if the role-group assigned to G contains a role in R

create positive rule between H_1 and H_2 with service=s

where r is a role, $ indicates the direction, R is a role-group and s is a service.

（対訳）
H_1に割当られたロールグループ内の各ロールｒに対して：
フォーム｛r$R:s｝内の各ステートメントに対して
H_2がクローズドの場合：／*H_2がロールｒを持つ場合はロールを生成*／H_2に割当られたロールグループがR内にロールを含む場合は、H_1とH_2との間のポジティブ規則を、sevice=sに対して、生成
そうでない場合は、H_2を含む全てのホストグループGに対して：Gに割当られたロールグループがR内にH_2を含む場合はH_1とH_2との間のポジティブ規則を、sevice=sに対して、生成
ここで、ｒはロールを表し、$は方向を表し、Rはロールグループを表し、ｓはサービスを表す。

【００６０】オープンホストグループHがクローズドホストグループｈを含む上述の例を用いて説明を続け、さらに、これもロールグループRを割当てられたもう一つのオープンホストグループH'が存在し、ロールグループRは、ホストが同一のロールグループを持つ他のホストに送信することを許可する（換言すれば、“Ｒ→Ｒ：ｓ１”なるフォームのMDLステートメントが存在する）ものと想定し、さらに、ロールグループCとRは、共通のロールは持たない（換言すれば、“Ｒ→Ｒ：ｓ１”なるフォームの定義は存在しない）ものと想定する。この場合は、規則生成アルゴリズム６００の過程３のみが適用され、過程３は、最初に、ｈに対するネガティブな規則を生成し、続いて、HとH'のペアに対するポジティブな規則を生成することで、結果として所望のセマンティクスを達成することとなる。

【００６１】上述のように、基本モデルコンパイラ４１０によって生成された中央ファイアウォールコンフィギュレーションファイル２５０Ａは、網１００内の各ゲートウェイ１２０、１５０に、各ゲートウェイインタフェース１２５、１５５に適合化された形で、配布する必要がある。セキュリティポリシーが守られることを確保するためには、全てのゲートウェイ内のそれらの間の全ての可能なルーティング経路に沿ってのペアのホストに関する全ての規則が含まれる必要がある。本発明によると、ルーティングプロトコルに関しての想定を設けることを回避するため、および、ルーティングの失敗に対する耐性を強化するために、“セーフ（安全）”戦略にて、中央ファイアウォールコンフィギュレーションファイル２５０Ａが各ゲートウェイ１２０、１５０上に複製される。

【００６２】上述のように、コンフィギュレーションファイルトポロジーアダプタ４２０は、規則の方向フィールドをセット（設定）する。ファイアウォールは、方向フィールドを以下のように用いる。つまり、ファイアウォールは、パケットがゲートウェイインタフェース１２５、１５５に入る方向をチェックし、その方向を、規則の方向フィールドと比較する。パケットがゲートウェイインタフェースから隣接ゾーンに出ることを試みている場合は、そのパケットは、その規則の方向が、IN（入り）あるいはBOTH（両方）に設定されている場合のみに許可される。同様にして、パケットは、隣接ゾーンからそのゲートウェイインタフェースに入ることを、その規則の方向が、OUT（出る）あるいはBOTH（両方）に設定されている場合のみに許可される。

【００６３】方向フィールドは、ロール（役割）の方向属性によっては含意（インプライ）されない。このことは、幾つかのホストグループはソースとして指定され、別の幾つかのホストグループは宛先として指定されることからも捉える（理解する）ことができる。他の情報が不在である場合は、規則の方向フィールドは、BOTHに設定することもできる。ただし、規則の方向フィールドは、できる限り厳密に設定するすべきである。そうする

ことで、ファイアウォール１２０、１５０は、ホストｈから到着したことを主張するパケットが、実際に、ｈに延びるゲートウェイインタフェース上にのみ出現することを保証することが可能となる。

【００６４】一般網トポロジーにおいては、ルーティングプロトコルに関する想定が設けられてない場合は、ソース宛先paid（経路識別子）は、ソースあるいは宛先がゲートウェイインタフェース１２５、１５５への隣接ゾーン内に位置する場合を除いて、パケットの方向に関しては、多くの情報は含蓄しない。このために、コンフィギュレーションファイルトポロジーアダプタ４２０は、図８に示すようなアルゴリズム８００を実現する。図８に示すように、最初に、中央コンフィギュレーションファイルが、各ゲートウェイインタフェース上に複製される。次に、アルゴリズム８００は、各ゲートウェイインタフェース、およびコンフィギュレーションファイル内の各規則に対して、以下の疑似コードを実現する：

【外２】

```
if the source is in the adjacent zone
        set direction to OUT
else if the destination is in the adjacent zone
        set direction to IN
else set direction to BOTH.
```

（対訳）

ソースが隣接ゾーン内に位置する場合は方向をOUTに設定

そうではなく、宛先が隣接ゾーン内に位置する場合は方向をINに設定

その他の場合は、方向をBOTHに設定

【００６５】不要な複製を回避するため、およびスプーフィングを防止するためには、ルーティング保証に関するある程度の知識が必要であり、この知識は、規則オプティマイザ内に備えることも、エンティティ関係モデル３００へのエクステンションの一部として設けることもできることに注意する。

【００６６】規則イラストレータ上述のように、視覚化／デバッギングツール２６０は、ファイアウォールコンフィギュレーションファイル２５０を、ホストグループの構造とファイアウォールを通過するサービス（パケット）の両方の視覚表現を提供する（視覚的に示す）グラフ表現に変換する。視覚化／デバッギングツール２６０は、単一のゲートウェイインタフェースの観点から見たときのセキュリティポリシーの視覚表現を生成する。視覚化／デバッギングツール２６０は、どのホストグループがゲートウェイインタフェース１２５）、１５５のどちら側に位置するか、およびそのゲートウェイインタフェースによって施行（実行）されるファイアウォール規則を表示する。

【００６７】視覚化／デバッギングツール２６０は、ファイアウォールマネージャ２００によるデバッギングタスクを楽にする。カラフルなグラフを見る方が、長い自動的に生成された、難解なファイアウォールフォーマットによるコンフィギュレーションファイルを、順番にシフトして調べるより容易であることはいうまでもない。ただし、視覚化／デバッギングツール２６０は、それ自体でも、有益である。例えば、視覚化／デバッギングツール２６０は、ファイアウォールコンフィギュレーションファイル２５０を読めるために、視覚化／デバッギングツール２６０を用いて、現存するコンフィギュレーションファイルをリバースエンジニアし、セキュリティポリシーを抽出することもできる。

【００６８】視覚化／デバッギングツール２６０は、ホストグループの構造を、コンテインメント（包含／部分集合、containment）と、インタセクション（共通／共通集合、intersection）という観点に分けて視覚化する。これは、あるホストグループＡに適用する規則は、そのIPアドレスがＡに属する全てのホストによって継承されるために重要である。

【００６９】ホストグループの構造は、そのノードがそのホストグループの名前にてラベリングされるグラフとして表示される。図９に示す実施例においては、２つのノードＡとＢとの間の太く黒いエッジ、例えば、線９１０、９２０は、一方のノードが他方のノードを包含することを示す。コンテインメント（包含）の方向（つまり、Ａ　BかB　Aかは）、どちらのノードが上であるかによって示される。点線の黒いエッジ、例えば、線９３０、９４０は、インタセクト（intersect、共通集合）である（換言すれば、重複するが、ただし、一方が他方を完全には包含しない）ホストグループを示す。

【００７０】最初に、ホストグループは、それらが位置するインタフェースのサイドによって、２つのカテゴリ９５０、９６０に分割される。一方のカテゴリは、“外側（outside”ゾーン９５０と呼ばれ、これは、典型的には、インターネットゾーン１１０を含む方であり、他方のカテゴリは、“内側（inside）”ゾーン９６０と呼ばれる。

【００７１】分割された網を視覚化するためには、_outおよび_inと呼ばれる２つの人口的な（アーティフィシャルな）ホストグループ９７０、９８０が導入され、これが、グラフ９００の中央に、２つのダイヤ形状のノード９７０、９８０として表示される（他のホストグループは楕円として示される）。内側ホストグループ９６０は、_inノード９８０から下向に成長する木として表示され、外側ホストグループ９５０は、_outノード９７０から上向に成長する木として表示される。こうして、内側ホストグループ９６０については、Ａはエッジ Ｂへの包含（インクルージョン、inclusion）エッジを含み、AがBより上である（Ａの方が_inノード９８０に近い）場合は、Ａ　Bの関係が成り立つ。外側ホスト

グループ９５０については、_outノード９７０に近い方のグループが、他方を包含する。木９５０、９６０は、その推移クロージャ（大小符号、transitive closure）が、ホストグループの包含関係に等しい最小包含関係を表す。その後、木の階層構造に従わない共通部分のエッジが追加される。最後に、ゾーンを表現するために、ノードにカラーが割り当られる。同一のゾーンに属するホストグループには、全て、同一のカラーが割り当られる。あるホストグループが、同時に、“外側”ゾーン９５０と、“内側”ゾーン９６０に属する場合は、そのホストグループは、さらに、２つの_inおよび_outのサブグループに分割され、各サブグループが別個に表示される。

【００７２】視覚化／デバッギングツール２６０は、ゲートウェイインタフェースを横断（通過）するサービスに対する規則のみを表示し、両方のエンドポイントがゲートウェイインタフェースの同一のサイドに位置するサービスを扱う規則は無視する。図９に示すように、これら規則は、ソースから宛先への有向エッジ（矢印）によって表現される。AからBへのエッジは、ファイアウォールが、ホストグループA（およびそのサブグループ）からホストグループB（およびそのサブグループ）にパス（通過）することを許すサービスを表す。様々な異なるサービスが、エッジをコード的に表現するカラーによって示される。例えば、全てのtcpサービスは、赤の矢印によって表され、全てのtelnetサービスは、青い矢印を用いて表示される。

【００７３】以上、本発明の様々な実施例およびバリエーションについて説明したが、これらは、単に、本発明の原理を解説するために示したものであり、当業者においては、本発明の範囲および精神から逸脱することなく、他の様々な修正が可能であると考えられる。

【００７４】例えば、モデルのフレームワーク（枠組）は、ファイアウォールの進化に合わせて容易に拡張することが可能である。例えば、継承（インヘリタンス）を介して新たな属性をオブジェクトに追加したり、元のモデル（規則）に違反することなく、全く新たなオブジェクトを追加することもできる。

【図面の簡単な説明】

【図１】本発明による一例としての網の環境を示す図である。

【図２】図１のファイアウォールマネージャの要素を示す図である。

【図３】図２のエンティティ関係（リレーションシップ）モデルの枠組を示す図である。

【図４】図２のベンダスペシフィックなコンパイラの略ブロック図である。

【図５】図４のベンダスペシフィックなコンパライによって遂行される各ゲートウェイインタフェースに対するコンフィギュレーションファイルの生成を示す流れ図である。

【図６Ａ】一体となって、図４の基本モデルコンパイラによって遂行される一例としての規則（ルール）生成アルゴリズムを記述する流れ図である。

【図６Ｂ】一体となって、図４の基本モデルコンパイラによって遂行される一例としての規則（ルール）生成アルゴリズムを記述する流れ図である。

【図７】図６Ａおよび図６Ｂの規則生成アルゴリズムによって遂行されるポジティブ（積極的）な規則のリストを生成するための一例としてのルーチンを示す図である。

【図８】規則生成アルゴリズムによって生成された規則の方向フィールドを設定するためのコンフィギュレーションファイルトポロジーアダプタによって遂行される一例としてのルーチンを示す図である。

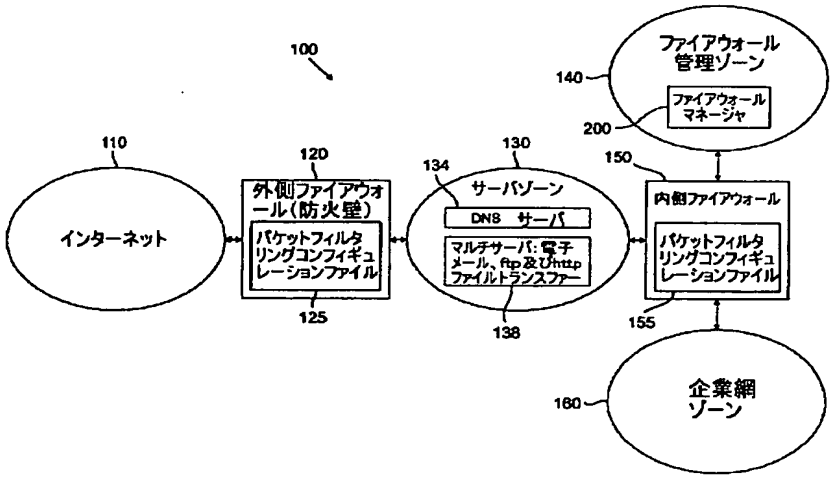
【図９】ホストグループの構造およびファイアウォールを通過するサービス（パケット）を視覚化するグラフ表現を示す図である。

【符号の説明】

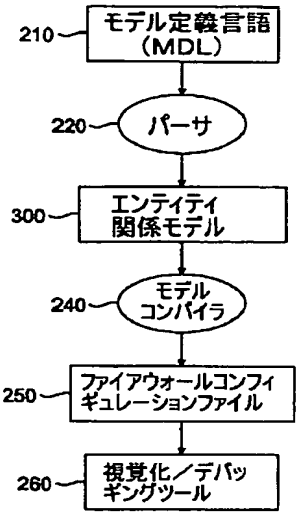
１００　網の環境
１１０　インターネット
１２０　外部ファイアウォール（防火壁）
１３０　サーバゾーン
１３４　DNSサーバ
１３８　マルチプルサーバ
１４０　ファイアウォール管理ゾーン
１２５、１５５　パケットフィルタリングコンフィギュレーションファイル（パケットフィルタリング規則ベースゲートウェイインタフェース）
１６０　企業網ゾーン
１５０　内部ファイアウォール
２００　ファイアウォールマネージャ
２１０　モデル定義言語（MDL）
２２０　パーサ（解析器）
２４０　モデルコンパイラ（ベンダスペシフィックコンパイラ）
２５０　ファイアウォールコンフィギュレーションファイル
２６０　視覚化／デバッギングツール
３００　エンティティ関係（リレーションシップ）モデル
３１０　ロール（役割）オブジェクト
３１５　能力（機能）
３２０　サービス
３３０　サービスグループ
３２５　ロールグループオブジェクト
３４０　ゾーンオブジェクト
３５０　ゲートウェイオブジェクト
３６０　ゲートウェイインタフェースオブジェクト
３７０　ホストグループオブジェクト

380　ホストオブジェクト
410　基本モデルコンパライ
420　コンフィギュレーションファイルトポロジーアダプタ

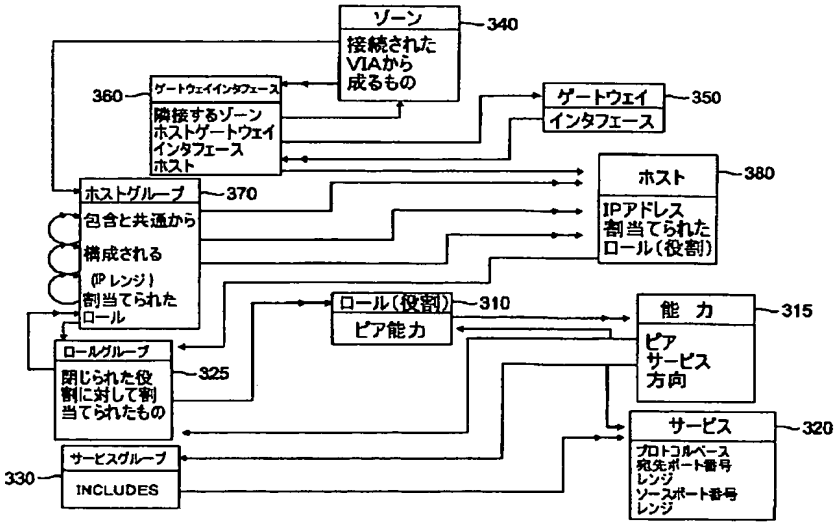
【図1】



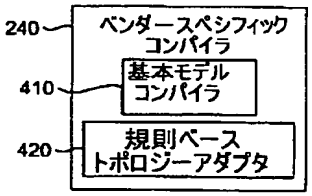
【図2】



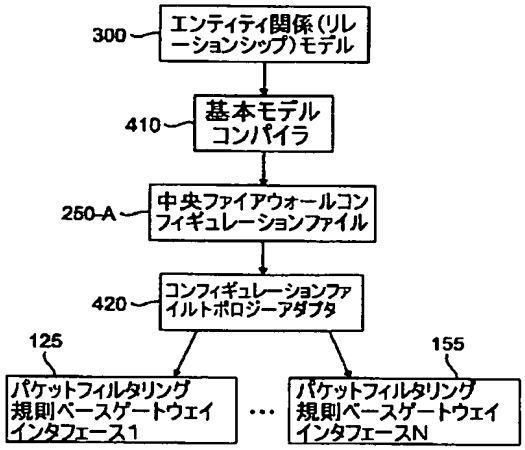
【図3】



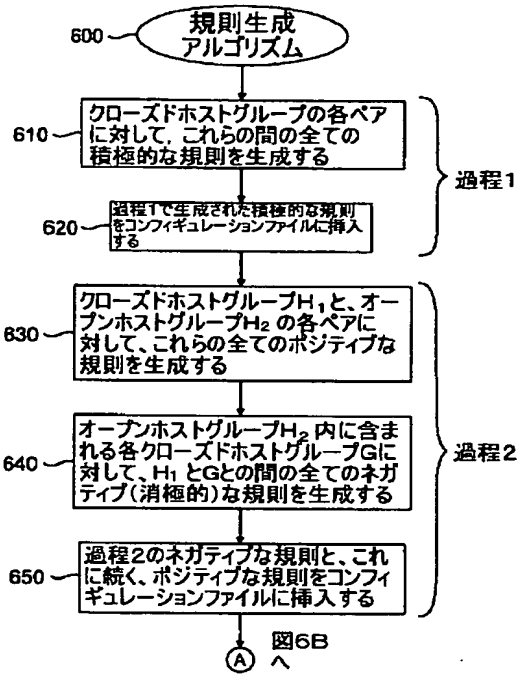
【図4】



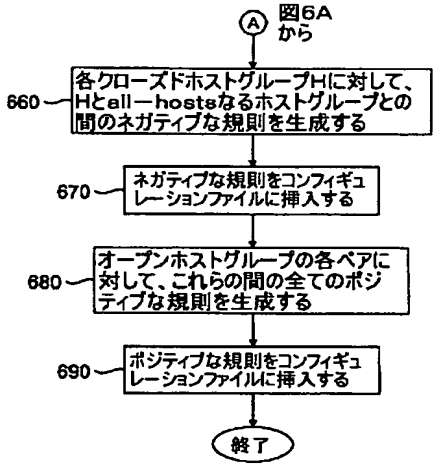
【図5】



【図6A】



【図6B】



【図7】

```
for each role r in the role-group assigned to H1:
    for each statement in the form: { r $ R : s}
        /* where r is a role, $ indicates the direction, R is a role-
        group and s is a service */
        if H2 is closed:  /* create a rule if H2 has a role, r */
            if the role-group assigned to H2 contains a role in R,
            then,
                create a positive rule between H1 and H2 with
                service = s
        otherwise, for all host-groupe G that contain H2:
            if the role-group assigned to G contains a role in R
                create positive rule between H1 and H2 with
                service = s
```

(対訳)

H1に割当られたロールグループ内の各ロールrに対して：

フォーム{r$R:s}内の各ステートメントに対して

／*rはロールを表し、$は方向を表し、Rはロールグループを表し、

sはサービスを表す*／

H2がクローズドの場合：／*H2がロールrを持つ場合はロールを生成*／

H2に割当られたロールグループがR内にロールを含む場合は、

H1とH2との間のポジティブ規則を、sevice=sに対して、生成

そうでない場合は、H2を含む全てのホストグループGに対して：

Gに割当られたロールグループがR内にH2を含む場合は

H1とH2との間のポジティブ規則を、sevice=sに対して、生成

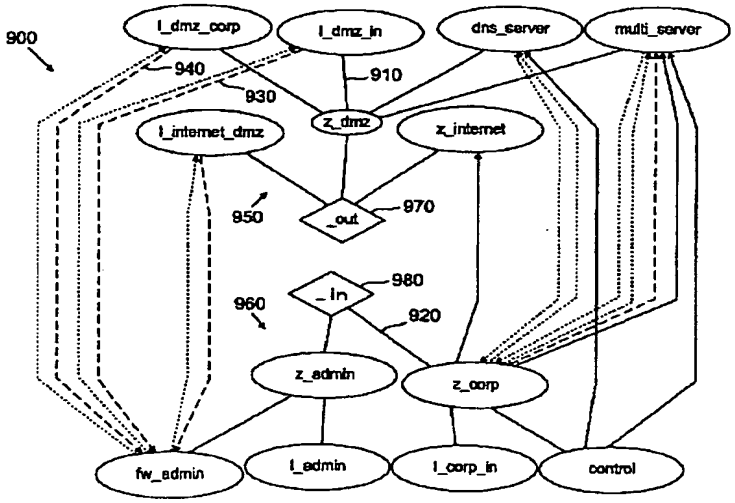
【図8】

800

```
Replicate the centralized configuration file to every gateway
interface
for each gateway interface
    for each rule in the configuration file
        if the source is in the adjacent zone
            set direction to OUT
        else if the destination is in the adjacent zone
            set direction to IN
        else set direction to BOTH
```

(対訳)
中央コンフィギュレーションファイルを各ゲートウェイインタフェース上に複製する
各ゲートウェイインタフェースに対して
コンフィギュレーションファイル内の各規則に対して
ソースが隣接ゾーン内に位置する場合は
方向を OUT に設定
そうではなく、宛先が隣接ゾーン内に位置する場合は
方向を IN に設定
その他の場合は、方向を BOTH に設定

【図9】



フロントページの続き

(51)Int.Cl.[7]	識別記号	F I	ﾃｰﾏｺｰﾄﾞ(参考)
H04L 12/26			
12/66			

(72)発明者 アレイン ジュレス メイヤー
アメリカ合衆国 10017 ニューヨーク，ニューヨーク，イー フォーティエイスストリート 230，アパートメント 6エフ

(72)発明者 アヴィシャイ ウール
アメリカ合衆国 07039 ニュージャーシィ，リビングストン，フェルスウッド ドライヴ 45